ELG5255 Applied Machine Learning Group Assignment 2

Group 9

1. Wael Mofreh Ismail Ali Elsharkawy – 300273116
2. Sameh Mohamed Ismail Deabes - 300273126

Part 1

Q1: Calculate mean

$$\mathbf{Mean}\ (\boldsymbol{\mu}) = \frac{\mathbf{x1+x2+\cdots\ldots\ldots+xn}}{\mathbf{n}}$$

Class 0

Mean (sepal length) = $\frac{5.1+5.0+4.8+5.0}{4}$ = 4.975

Mean (sepal width) = $\frac{3.4+3.4+3.0+3.3}{4}$ = 3.275

Mean (petal length) = $\frac{1.5+1.5+1.4+1.4}{4}$ = 1.45

Mean (petal width) = $\frac{0.2+0.2+0.1+0.2}{4}$ = 0.175

Class 1

Mean (sepal length) = $\frac{4.9+5.7+5.4+5.6}{4}$ = 5.4

Mean (sepal width) = $\frac{2.4+3.0+3.0+2.5}{4}$ = 2.725

Mean (petal length) = $\frac{3.3+4.2+4.5+3.9}{4}$ = 3.975

Mean (petal width) = $\frac{1.0+1.2+1.5+1.1}{4}$ = 1.2

Class 2

Mean (sepal length) = $\frac{6.5+7.7}{2}$ = **7.1**

Mean (sepal width) = $\frac{3.0+2.6}{2}$ = **2.8**

Mean (petal length) = $\frac{5.8+6.9}{2}$ = **6.35**

Mean (petal width) = $\frac{2.2+2.3}{2}$ = **2.25**

Q2: Calculate variance

$$\boldsymbol{variance(\sigma^2) =} \qquad \frac{\sum(x_i - \bar{x})^2}{n-1}$$

Class 0

variance (sepal length) = $\frac{(5.1 - 4.975)^2+(5.0-4.975)^2+(4.8- 4.975)^2+(5.0-4.975)^2}{4-1} = 0.015833$

variance (sepal width) = $\frac{(3.4 - 3.275)^2+(3.4-3.275)^2+(3.0- 3.275)^2+(3.3-3.275)^2}{4-1} = 0.035833$

variance (petal length) = $\frac{(1.5 - 1.45)^2+(1.5-1.45)^2+(1.4- 1.45)^2+(1.4-1.45)^2}{4-1} = 0.003333$

variance (petal width) = $\frac{(0.2 - 0.175)^2+(0.1-0.175)^2+(0.2- 0.175)^2+(0.2-0.175)^2}{4-1} = 0.0025$

Class 1

variance (sepal length) = $\frac{(4.9 - 5.4)^2+(5.7 - 5.4)^2+(5.4 - 5.4)^2+(5.6 - 5.4)^2}{4-1} = 0.126666$

variance (sepal width) = $\frac{(2.4 - 2.725)^2+(3.0-2.725)^2+(3.0 - 2.725)^2+(2.5 - 2.725)^2}{4-1} = 0.1025$

variance (petal length) = $\frac{(3.3 - 3.975)^2+(4.2 - 3.975)^2+(4.5 - 3.975)^2+(3.9 - 3.975)^2}{4-1} = 0.2625$

variance (petal width) = $\frac{(1.0- 1.2)^2+(1.2 - 1.2)^2+(1.5-1.2)^2+(1.1 - 1.2)^2}{4-1} = 0.046666$

Class 2

variance (sepal length) = $\frac{(6.5-7.1)^2+(7.7-7.1)^2}{2-1} = 0.72$

variance (sepal width) = $\frac{(3.0-2.8)^2+(2.6-2.8)^2}{2-1} = 0.08$

variance (petal length) = $\frac{(5.8-6.35)^2+(6.9-6.35)^2}{2-1} = 0.605$

variance (petal width) = $\frac{(2.2-2.25)^2+(2.3-2.25)^2}{2-1} = 0.005$

Q3-

sepal length	sepal width	petal length	petal width	Label
5.7	2.8	4.5	1.3	
5.4	3.9	1.3	0.4	

Posterior (*class*) = *Piror* × *likelihood*

For Row1

***likelihood*:**

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

Likelihood for point sepal length (5.7) for class 0 (μ = 4.975, $\boldsymbol{\sigma^2}$= 0.015833**)**

$$P(5.7) = \frac{1}{\sqrt{2\times\pi\times 0.015833}}\, e^{\frac{-(5.7-4.975)^2}{2\times 0.015833}} = 1.96001653\times 10^{-7}$$

Likelihood for point sepal width (2.8) for class 0 (μ = 3.275, $\boldsymbol{\sigma^2}$= 0.035833**)**

$$P(2.8) = \frac{1}{\sqrt{2\times\pi\times 0.035833}}\, e^{\frac{-(2.8-3.275)^2}{2\times 0.035833}} = 0.09046605131$$

Likelihood for point petal length (4.5) for class 0 (μ = 1.45, $\boldsymbol{\sigma^2}$= 0.003333**)**

$$P(4.5) = \frac{1}{\sqrt{2\times\pi\times 0.003333}}\, e^{\frac{-(4.5-1.45)^2}{2\times 0.003333}} = 0$$

Likelihood for point petal width (1.3) for class 0 (μ = 0.175, $\boldsymbol{\sigma^2}$= 0.0025**)**

$$P(1.3) = \frac{1}{\sqrt{2\times \pi \times 0.0025}}\ e^{\frac{-(1.3-0.175)^2}{2\times 0.0025}} = 9.35727363\times\ 10^{-110}$$

$Piror$ for class $0 = \frac{4}{10}$

Posterior ($class$ 0) = $\frac{4}{10}\ \times 1.96001653\times\ 10^{-7}\times\ 0.09046605131\times\ 0\times\ 9.35727363\times\ 10^{-110}$ = **0**

Likelihood for point sepal length (5.7) for class 1 (μ = 5.4, σ^2 = 0.126666)

$$P(5.7) = \frac{1}{\sqrt{2\times \pi \times 0.126666}}\ e^{\frac{-(5.7-5.4)^2}{2\times 0.126666}} = 0.7857608681$$

Likelihood for point sepal width (2.8) for class 1 (μ = 2.725, σ^2 = 0.1025)

$$P(2.8) = \frac{1}{\sqrt{2\times \pi \times 0.1025}}\ e^{\frac{-(2.8-2.725)^2}{2\times 0.1025}} = 1.212359769$$

Likelihood for point petal length (4.5) for class 1 (μ = 3.975, σ^2 = 0.2625)

$$P(4.5) = \frac{1}{\sqrt{2\times \pi \times 0.2625}}\ e^{\frac{-(4.5-3.975)^2}{2\times 0.2625}} = 0.4606178978$$

Likelihood for point petal width (1.3) for class 1 (μ = 1.2, σ^2 = 0.046666)

$$P(1.3) = \frac{1}{\sqrt{2\times \pi \times 0.046666}}\ e^{\frac{-(1.3-1.2)^2}{2\times 0.046666}} = 1.659119093$$

$Piror$ for class $1 = \frac{4}{10}$

Posterior ($class$ 1) = $\frac{4}{10}\ \times 0.7857608681\times\ 1.212359769\times 0.4606178978\times 1.659119093$ = **0.2912059701**

Likelihood for point sepal length (5.7) for class 2 (μ = 7.1, σ^2= 0.72)

$$P(5.7) = \frac{1}{\sqrt{2\times \pi \times 0.72}} e^{\frac{-(5.7-7.1)^2}{2\times 0.72}} = 0.1205371095$$

Likelihood for point sepal width (2.8) for class 2 (μ = 2.8, σ^2= 0.08)

$$P(2.8) = \frac{1}{\sqrt{2\times \pi \times 0.08}} e^{\frac{-(2.8-2.8)^2}{2\times 0.08}} = 1.410473959$$

Likelihood for point petal length (4.5) for class 2 (μ = 6.35, σ^2= 0.605)

$$P(4.5) = \frac{1}{\sqrt{2\times \pi \times 0.605}} e^{\frac{-(4.5-6.35)^2}{2\times 0.605}} = 0.0303127301$$

Likelihood for point petal width (1.3) for class 2 (μ = 2.25, σ^2= 0.005)

$$P(1.3) = \frac{1}{\sqrt{2\times \pi \times 0.005}} e^{\frac{-(1.3-2.25)^2}{2\times 0.005}} = 3.60037777 \times 10^{-39}$$

$Piror$ for class 2 $= \frac{2}{10}$

Posterior ($class$ 2) = $\frac{2}{10} \times 0.1205371095\times 1.410473959\times 0.0303127301\times 3.60037777 \times 10^{-39} = \mathbf{3.710983 \times 10^{-42}}$

So as conclusion for row 1

- **Posterior ($class$ 0) = 0**
- **Posterior ($class$ 1) = 0.2912059701**
- **Posterior ($class$ 2) = $\mathbf{3.710983 \times 10^{-42}}$**

So our prediction will be class 1 for row 1

For Row2

$likelihood$:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

Likelihood for point sepal length (5.4) for class 0 (μ = 4.975, σ^2= 0.015833)

$$P(5.4) = \frac{1}{\sqrt{2 \times \pi \times 0.015833}} e^{\frac{-(5.4-4.975)^2}{2 \times 0.015833}} = 0.01056533909$$

Likelihood for point sepal width (3.9) for class 0 (μ = 3.275, σ^2= 0.035833)

$$P(3.9) = \frac{1}{\sqrt{2 \times \pi \times 0.035833}} e^{\frac{-(3.9-3.275)^2}{2 \times 0.035833}} = 0.009048759334$$

Likelihood for point petal length (1.3) for class 0 (μ = 1.45, σ^2= 0.003333)

$$P(1.3) = \frac{1}{\sqrt{2 \times \pi \times 0.003333}} e^{\frac{-(1.3-1.45)^2}{2 \times 0.003333}} = 0.2363752186$$

Likelihood for point petal width (0.4) for class 0 (μ = 0.175, σ^2= 0.0025)

$$P(0.4) = \frac{1}{\sqrt{2 \times \pi \times \mathbf{0.0025}}} e^{\frac{-(0.4-0.175)^2}{2 \times 0.0025}} = 3.19674822 \times 10^{-4}$$

Piror for class 0 = $\frac{\mathbf{4}}{\mathbf{10}}$

Posterior (*class* **0**) = $\frac{\mathbf{4}}{\mathbf{10}} \times 0.01056533909 \times 0.009048759334 \times 0.2363752186 \times 3.19674822 \times 10^{-4} = \mathbf{2.88963404 \times 10^{-9}}$

Likelihood for point sepal length (5.4) for class 1 (μ = 5.4, σ^2= 0.126666)

$$P(5.4) = \frac{1}{\sqrt{2 \times \pi \times \mathbf{0.126666}}} e^{\frac{-(5.4-5.4)^2}{2 \times 0.126666}} = 1.120933988$$

Likelihood for point sepal width (3.9) for class 1 (μ = 2.725, σ^2= 0.1025)

$$P(3.9) = \frac{1}{\sqrt{2 \times \pi \times \mathbf{0.1025}}} e^{\frac{-(3.9-2.725)^2}{2 \times 0.1025}} = 0.001481428594$$

Likelihood for point petal length (1.3) for class 1 (μ = 3.975, σ^2= 0.2625)

$$P(1.3) = \frac{1}{\sqrt{2 \times \pi \times \mathbf{0.2625}}} e^{\frac{-(1.3-3.975)^2}{2 \times 0.2625}} = 9.37594448 \times 10^{-7}$$

Likelihood for point petal width (0.4) for class 1 (μ = 1.2, σ^2= 0.046666)

$$P(0.4) = \frac{1}{\sqrt{2 \times \pi \times \mathbf{0.046666}}} e^{\frac{-(0.4-1.2)^2}{2 \times 0.046666}} = 0.001942441416$$

Piror for class 1 $= \frac{4}{10}$

Posterior (*class* 1) $= \frac{4}{10} \times 1.120933988 \times 0.001481428594 \times 9.37594448 \times 10^{-7} \times 0.001942441416 = \mathbf{1.20971679 \times 10^{-12}}$

Likelihood for point sepal length (5.4) for class 2 (μ = 7.1, σ^2= 0.72)

$$P(5.4) = \frac{1}{\sqrt{2\times \pi \times 0.72}} e^{\frac{-(5.4-7.1)^2}{2\times 0.72}} = 0.06318862715$$

Likelihood for point sepal width (3.9) for class 2 (μ = 2.8, σ^2= 0.08)

$$P(3.9) = \frac{1}{\sqrt{2\times \pi \times 0.08}} e^{\frac{-(3.9-2.8)^2}{2\times 0.08}} = 7.32846559 \times 10^{-4}$$

Likelihood for point petal length (1.3) for class 2 (μ = 6.35, σ^2= 0.605)

$$P(1.3) = \frac{1}{\sqrt{2\times \pi \times 0.605}} e^{\frac{-(1.3-6.35)^2}{2\times 0.605}} = 3.60286559 \times 10^{-10}$$

Likelihood for point petal width (0.4) for class 2 (μ = 2.25, σ^2= 0.005)

$$P(0.4) = \frac{1}{\sqrt{2\times \pi \times 0.005}} e^{\frac{-(0.4-2.25)^2}{2\times 0.005}} = 1.30058467 \times 10^{-148}$$

Piror for class 2 $= \frac{2}{10}$

Posterior (*class* 2) $= \frac{2}{10} \times 0.06318862715 \times 7.32846559 \times 10^{-4} \times 3.60286559 \times 10^{-10} \times 1.30058467 \times 10^{-148} = 4.33978945 \times \mathbf{10^{-163}}$

So as conclusion for row 2

- **Posterior (*class* 0) = $\mathbf{2.88963404 \times 10^{-9}}$**
- **Posterior (*class* 1) = $\mathbf{1.20971679 \times 10^{-12}}$**
- **Posterior (*class* 2) = $\mathbf{4.33978945 \times 10^{-163}}$**

<u>So prediction will be class 0 for row 2</u>

Final Table 2

sepal length	sepal width	petal length	petal width	Label
5.7	2.8	4.5	1.3	1
5.4	3.9	1.3	0.4	0

Part 2

1. **We load Iris Dataset and prepare it to work on it and make function to use it in presentation**
2. **Dropping 2 future the petal length and petal width feature from 4 to work on 2D future**

```
# Load the Iris dataset
iris = datasets.load_iris()
#Drop the petal length and petal width features to form a 2D Iris dataset
X, y = iris.data[:, [0, 1]], iris.target
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)
```

3. **Apply Naïve Bayes Classifier to test and training data**
   we make the model and make prediction using training data and fitting it on test data

```
nb = GaussianNB()
nb.fit(X_train, y_train)
yPred = nb.predict(X_test)

ytrPred = nb.predict(X_train)
```

```
# Model Accuracy, how often is the classifier correct?
print("Naive Bayes Accuracy of train set:",metrics.accuracy_score(y_train, ytrPred))
print("Naive Bayes Accuracy of test set:",metrics.accuracy_score(y_test, yPred))
```

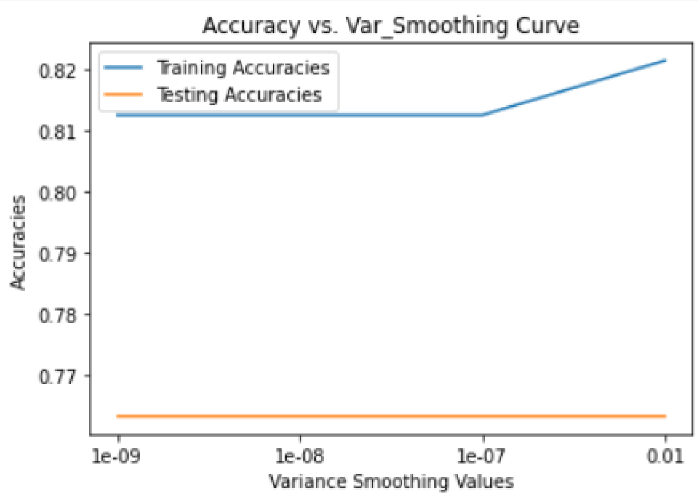
Output and the accuracy

```
Naive Bayes Accuracy of train set: 0.8125
Naive Bayes Accuracy of test set: 0.7631578947368421
```

4. **Try var_smoothing as parameter to naïve bayes classifier and find the change in accuracy (1e-9, 1e-8, 1e-7 ,1e-2). Plot accuracy vs var_smoothing curve for training and testing set**

```
#4. Tune hyperparameters of Naive Bayes Classifier (i.e., var_smoothing). Try var_smoothing as
#1e-9, 1e-8, 1e-7. Plot accuracy vs var_smoothing curve for training and testing set. (10 marks)
var_smoothing_vals = [1e-9,1e-8,1e-7,1e-2]
train_accuracies=[]
test_accuracies=[]

#model 1 that with var_smoothing = 1e-9
nb1 = GaussianNB(var_smoothing=1e-9)
nb1.fit(X_train, y_train)
yPred1 = nb1.predict(X_test)
ytrPred1 = nb1.predict(X_train)
# get model train and test Accuracies
train_accuracy1 = metrics.accuracy_score(y_train, ytrPred1)
test_accuracy1 = metrics.accuracy_score(y_test, yPred1)
# append accuricies to accuracy array
train_accuracies.append(train_accuracy1)
test_accuracies.append(test_accuracy1)
```



the accuracy does not change when we change var-smoothing with these values (1e-9, 1e-8, 1e-7) and when we minimize the value to 1e-2 the accuracy of train increases and the accuracy of testing still the same.

Q5 : Develop Risk-based Bayesian Decision Theory Classifier (RBDTC)

- The classifier Class inherit BaseEstimator and ClassifierMinxin in sklearn.
- The classifier check inputs Using (check_X_y(X,y)) and model status using (check_is_fitted(self)).
- __init__ function is able to take risk matrix and any kind of base estimator as inputs.
- fit function is able to take training dataset (X and y) as input.
- predict_proba function take only testing X as input. And the output predict_proba is a matrix.
- predict function is able to output text format y.

Code:

```
class BayesianDecisionTheoryClassifier(BaseEstimator, ClassifierMixin):
    def __init__(self, estimator, utilityMat):
        self.estimator = estimator
        self.utilityMat = utilityMat

    def fit(self, X, y):
      # check inputs
        check_X_y(X,y)
        self.classes_ = np.unique(y)
        self.estimator_ = clone(self.estimator).fit(X, y)
        return self

    def predict_proba(self, X):
        check_is_fitted(self)
        prob = self.estimator_.predict_proba(X)
        probList = [(prob * self.utilityMat[index]).sum(axis=1).reshape((-1, 1))
                    for index, c in enumerate(self.classes_)]
        prob = np.hstack(probList)
        return prob
    # predict be able to output text format y by making outputTxt = True
    def predict(self, X, outputTxt=False):
      pred = self.predict_proba(X).argmin(axis=1)
      if outputTxt:
          new_y = []
          for i,value in enumerate(pred):
            if value == 0:
              new_y.append('Setosa')
            elif value == 1:
              new_y.append('Versicolor')
            else:
              new_y.append('Virginica')
          pred= new_y
          return pred
      return self.classes_[pred]
```

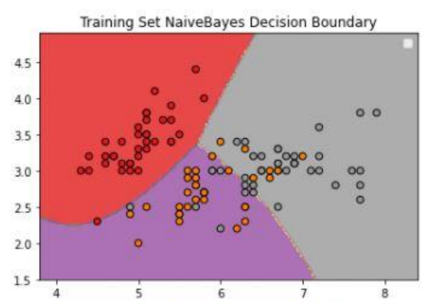
Q6- Apply Risk-based Bayesian Decision Theory Classifier which takes Naïve Bayes Classifier as base estimator and uses Table 3 as risk matrix

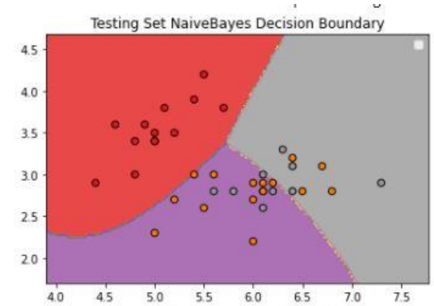
```python
utilityMat = np.array([
    [-10, -5, -5],
    [-5, -10, -5],
    [-5, -5, -100],
])

bdtc = BayesianDecisionTheoryClassifier(nb, utilityMat)
bdtc.fit(X_train, y_train)
yPred = bdtc.predict(X_test)
ytrPred = bdtc.predict(X_train)
```

Q7- Decision boundary and precision, recall and accuracy for training and testing.

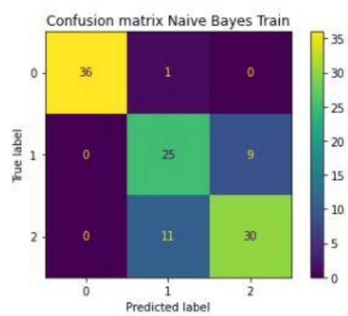
For NB:

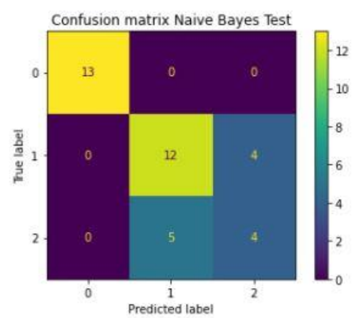




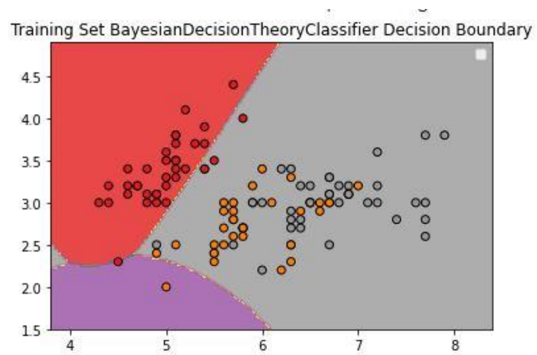
	precision	recall	f1-score	support
0	1.00	0.97	0.99	37
1	0.68	0.74	0.70	34
2	0.77	0.73	0.75	41
accuracy			0.81	112
macro avg	0.81	0.81	0.81	112
weighted avg	0.82	0.81	0.81	112

	precision	recall	f1-score	support
0	1.00	1.00	1.00	13
1	0.71	0.75	0.73	16
2	0.50	0.44	0.47	9
accuracy			0.76	38
macro avg	0.74	0.73	0.73	38
weighted avg	0.76	0.76	0.76	38

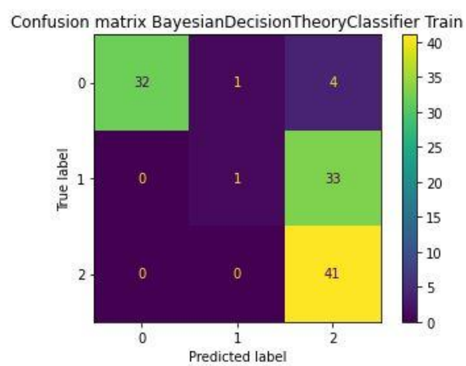


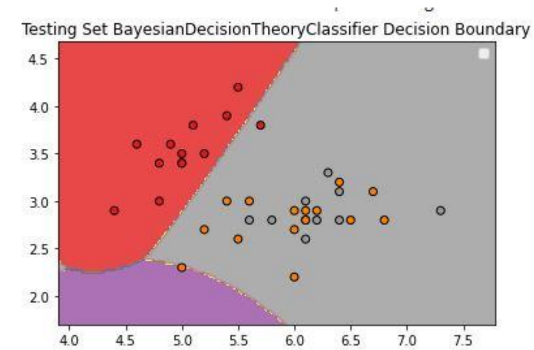


For RBDTC :

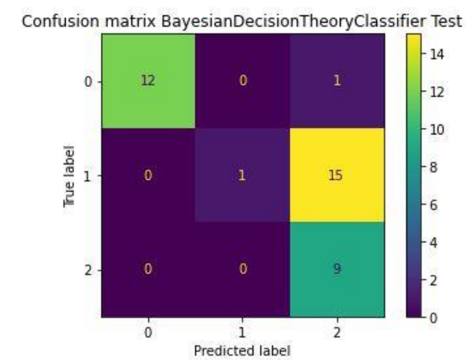


	precision	recall	f1-score	support
0	1.00	0.86	0.93	37
1	0.50	0.03	0.06	34
2	0.53	1.00	0.69	41
accuracy			0.66	112
macro avg	0.68	0.63	0.56	112
weighted avg	0.67	0.66	0.58	112





	precision	recall	f1-score	support
0	1.00	0.92	0.96	13
1	1.00	0.06	0.12	16
2	0.36	1.00	0.53	9
accuracy			0.58	38
macro avg	0.79	0.66	0.54	38
weighted avg	0.85	0.58	0.50	38



Q8: Compare and analysis the performance between NB and RBDTC regarding to their decision boundary, precision, recall, and accuracy

		Accuracy	Precision	Recall
NB	**Train**	**0.81**	**0.82**	**0.81**
	Test	**0.76**	**0.76**	**0.76**
RBDTC	**Train**	**0.66**	**0.67**	**0.66**
	Test	**0.58**	**0.85**	**0.58**

According to this table we find that NB is better than RBDTC in accuracies in training and testing. And for RBDTC classifier we found that Precision higher than NB as in Risk metrics we have class 2 with the value of 100 . and for Recall NB is better than RBDTC.

Conclusion:

As a part of this assignment we have learnt to

- Calculate mean and variance values of each class of each feature.
- Calculate the posterior probabilities for each sample for each class to predict classes.
- changing hyperparameters of Naive Bayes Classifier and look for accuracy.
- Develop Risk-based Bayesian Decision Theory Classifier class.
- using risk matrix to give different priority to a class
- Compare and analysis the performance between NB and RBDTC.